

soft-decision Dekodierung

- **optimaler** sum-product Algorithmus,
auch message-passing, belief-propagation (BP)
Basis: Wahrscheinlichkeiten oder Zuverlässigkeiten;

Zuverlässigkeit der Summe zweier Signalwerte:

$$\begin{aligned}
 L(y_1 \oplus y_2) &= \ln \frac{1 + e^{y_1} e^{y_2}}{e^{y_1} + e^{y_2}} \\
 &= \text{sign } y_1 \cdot \text{sign } y_2 \cdot \min\{|y_1|, |y_2|\} \\
 &\quad \underbrace{+ \ln(1 + e^{-|y_1+y_2|}) - \ln(1 + e^{-|y_1-y_2|})}_{\text{Lookup-Tabelle, stückweise lineare Abbildung, ...}}
 \end{aligned}$$

- **sub-optimaler** min-sum Algorithmus

$$\begin{aligned}
 L(y_1 \oplus y_2) &\approx \text{sign } y_1 \cdot \text{sign } y_2 \cdot \min\{|y_1|, |y_2|\} \\
 \longrightarrow L(y_1 \oplus y_2 \oplus \dots) &\approx \prod_j \text{sign } y_j \cdot \min_j |y_j|
 \end{aligned}$$

- **quasi-optimale** min-sum Algorithmen zur Wiederherstellung des Leistungsverlustes von min-sum zu sum-product:

normalized: $\alpha = 1.25 \dots 2.00$ (abh. von H -Konstruktion und γ)

$$L(y_1 \oplus y_2) \approx \frac{1}{\alpha} \cdot \text{sign } y_1 \cdot \text{sign } y_2 \cdot \min\{|y_1|, |y_2|\}$$

offset: $\beta = 0.15 \dots 0.20$

$$L(y_1 \oplus y_2) \approx \text{sign } y_1 \cdot \text{sign } y_2 \cdot \max\{\min\{|y_1|, |y_2|\} - \beta, 0\}$$

self-corrected (2008):

bei VZ Wechsel zwischen zwei Iterationen Nullsetzung (Auslöschung) \rightarrow Experimentierbaustein siehe web Kanalkodierung

... weitere Modifizierungen

Zur Einordnung der Algorithmen:

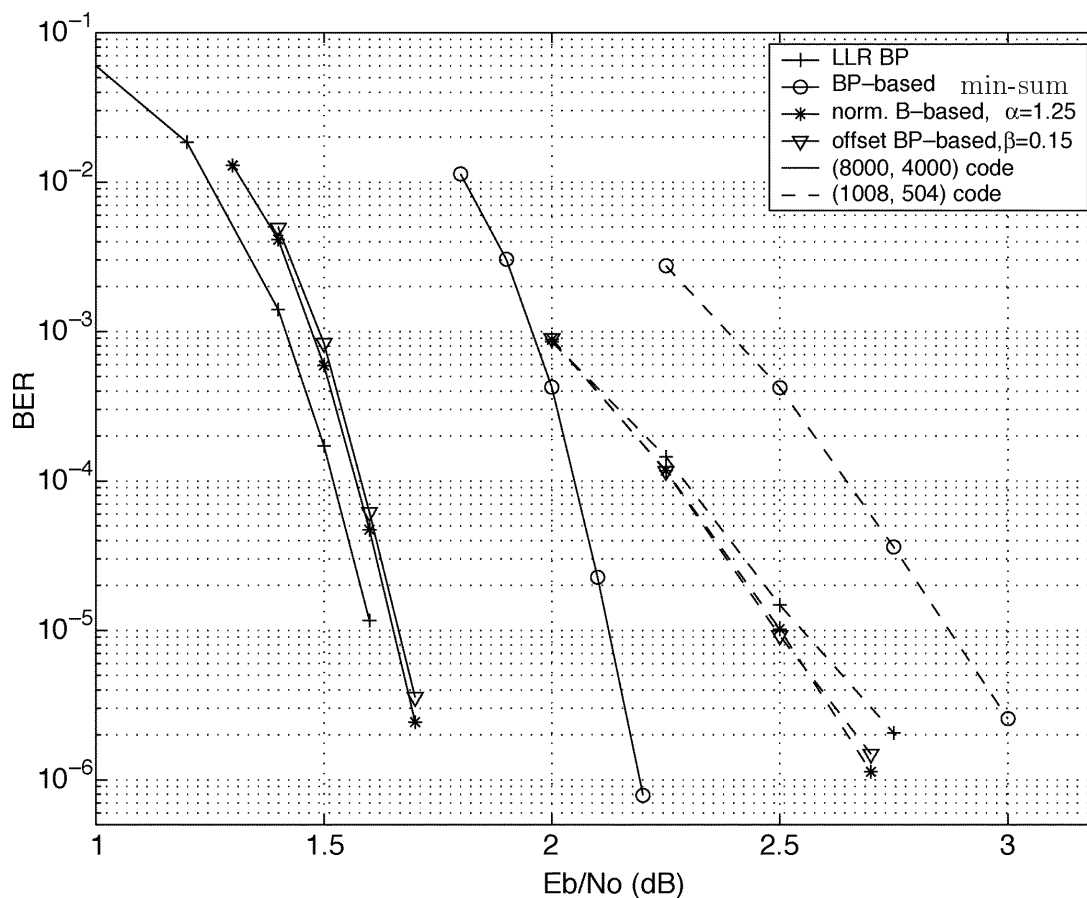


Fig. 2. Bit-error performances of an (8000,4000) and a (1008,504) LDPC code under two improved BP-based algorithms ($it_{\max} = 100$).

J. Chen, A. Dholakia, E. Eleftheriou, M. P.C. Fossorier. *Reduced-Complexity Decoding of LDPC Codes*. IEEE Trans. on Comm., Vol. 53, No. 8, Aug. 2005, S. 1293

→ bei kurzen/mittleren Kodewortlängen ist sum-product wegen vorhandener Korrelationen zwischen den Nachrichten nicht optimal

→ bei großen Kodewortlängen liegt Abstand zwischen quasi-optimalen und optimalem Algorithmus bei 0,5 ... 0,6 dB

soft-decision Umsetzung am Beispiel des

- optimalen **sum-product** Algorithmus

$$a \in \{0, 1\}^n \rightarrow a_M \in \{+1, -1\}^n \rightarrow b_M \in \mathbb{R}^n \rightarrow b \in \mathbb{R}[0, 1]^n$$

$$u_j \in \{0, 1\} \rightarrow x_j \in \{+1, -1\} \rightarrow y_j \in \mathbb{R} \rightarrow p(y_j) \in \mathbb{R}[0, 1]$$

$$y_j = x_j + z_j \text{ mit } z_j = \mathcal{Z}\mathcal{Z}_{NV[0, \sigma^2]}, \quad \sigma = \frac{1}{\sqrt{2 R 10^{\frac{E_b}{N_0} [dB]/10}}}$$

(AWGN-Modell mit Kanalzuverlässigkeit $L_c = \frac{2}{\sigma^2}$;

Algorithmus erfordert Kenntnis von L_c !)

und damit

$$p(y_j) = \frac{1}{1 + e^{-\frac{2}{\sigma^2} y_j}} \quad (j = 1, 2, \dots, n)$$

und

$$y_{h,j} = \begin{cases} 0 & p(y_j) \geq 0.5 \\ 1 & \text{sonst} \end{cases}$$

→ **Algorithmus** auf Basis von Wahrscheinlichkeiten
[probability domain]

Aufbau von 6 Matrizen:

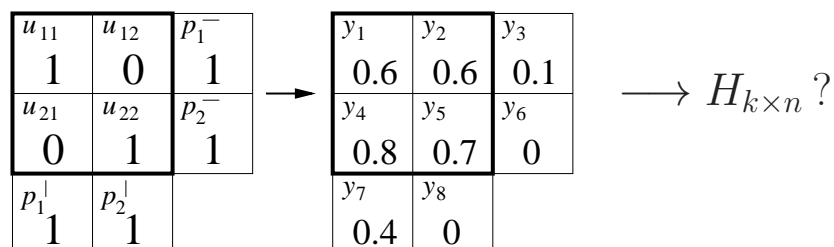
$$Q_{k \times n}^0, Q_{k \times n}^1, \Delta Q_{k \times n}, \Delta R_{k \times n}, R_{k \times n}^0, R_{k \times n}^1$$

→ Algorithmus in Anlehnung an MacKay/Neal [27], 1996

$$\forall i, j \wedge h_{ij} = 1 :$$

	1. Iteration	>1. Iteration
vertikal	$q_{ij}^0 = p(y_j)$ $q_{ij}^1 = 1 - q_{ij}^0$	$q_{ij}^0 = p(y_j) \prod_{i' \in N_j \setminus i} r_{i'j}^0 = q_j^0 / r_{ij}^0$ $q_{ij}^1 = (1 - p(y_j)) \prod_{i' \in N_j \setminus i} r_{i'j}^1 = q_j^1 / r_{ij}^1$ Skalieren: $q_{ij}^0 + q_{ij}^1 = 1$
horizontal	$\delta q_{ij} = q_{ij}^0 - q_{ij}^1$ $\delta r_{ij} = \prod_{j' \in K_i \setminus j} \delta q_{ij'}$ $r_{ij}^0 = \frac{1}{2}(1 + \delta r_{ij})$ und $r_{ij}^1 = \frac{1}{2}(1 - \delta r_{ij}) = 1 - r_{ij}^0$	
Auswertung	$q_j^0 = p(y_j) \prod_{i \in N_j} r_{ij}^0$ und $q_j^1 = (1 - p(y_j)) \prod_{i \in N_j} r_{ij}^1$ Skalieren, so dass $q_j^0 + q_j^1 = 1$ mit $q_j^0 = q_j^0 \cdot \frac{1}{q_j^0 + q_j^1}$ und $q_j^1 = 1 - q_j^0$ $s = H \cdot b_{korr,h}^T = H \cdot (q_{h,1}^0, q_{h,2}^0, \dots, q_{h,n}^0)^T = \mathbf{0} ?$ ja: stop bzw. τ_{max} erreicht nein: weiter mit vertikalen Schritt	

Beispiel $p(y_j | u = 0) \in \mathbb{R}[0, 1] :$



Weiter: $y_j \sim \ln \frac{p(y_j|u=0)}{p(y_j|u=1)} = \ln \frac{p(y_j|u=0)}{1-p(y_j|u=0)}$

- sub-optimalen **min-sum** Algorithmus¹³

$$a \in \{0, 1\}^n \rightarrow a_M \in \{+1, -1\}^n \rightarrow b_M \in \mathbb{R}^n \rightarrow b \in \mathbb{R}^n$$

→ **Algorithmus** auf Basis von Zuverlässigkeiten [log domain]

$$L(y_1 \oplus y_2 \oplus \dots) \approx \prod_j \text{sign } y_j \cdot \min_j |y_j|$$

Aufbau von 2 Matrizen: $Q_{k \times n}, R_{k \times n}$

$$\forall i, j \wedge h_{ij} = 1 :$$

	1. Iteration	> 1. Iteration
vertikal	$L(q_{ij}) = y_j$	$L(q_{ij}) = y_j + \sum_{i' \in N_j \setminus i} L_e(r_{i'j}) = L(\hat{u}_j) - L_e(r_{ij})$
horizontal	$L_e(r_{ij}) = \prod_{j' \in K_i \setminus j} \text{sign } L(q_{ij'}) \cdot \min_{j' \in K_i \setminus j} L(q_{ij'}) $	
Auswertung	$L(\hat{u}_j) = y_j + \sum_{i \in N_j} L_e(r_{ij})$ $\hat{u}_j = \begin{cases} 0 & L(\hat{u}_j) \geq 0 \\ 1 & \text{sonst} \end{cases} \quad (j = 1, 2, \dots, n)$ $s = H \cdot b_{\text{korrr},h}^T = H \cdot (\hat{u}_1, \hat{u}_2, \dots, \hat{u}_n)^T = \mathbf{0} ?$ <p>ja: stop bzw. τ_{max} erreicht nein: weiter mit vertikalem Schritt</p>	

Beispiel (Fortsetzung von Beispiel S. 150), $y_j \in \mathbb{R}$

$$b = (+0.4 \quad +0.4 \quad -2.2 \quad +1.4 \quad +0.8 \quad -7.0 \quad -0.4 \quad -7.0)$$

(Auch möglich: Auslöschungskorrektur!

$$b_{\text{ausl}} = (-1 \quad +1 \quad \mathbf{0} \quad \mathbf{0} \quad -1 \quad -1 \quad \mathbf{0} \quad -1); y_j \in \left\{ \underbrace{+1, -1}_{u_j = \frac{1 - \text{sign } y_j}{2}}, \mathbf{0} \right\}$$

¹³horizontale Umsetzung für quasi-optimale Algorithmen: s. S. 147

Vorteile LDPC gegenüber Turbokode?

- Parallelisierung der iterativen Dekodierung möglich; sehr geringe Dekodierungskomplexität
(Nachteil LDPC: teils hohe Kodierungskomplexität, abhängig von H -Konstruktion)
- „gute“ LDPC-Kodes entdecken nahezu alle nichtkorrigierbaren Empfangsfolgen
→ LDPC damit auch für HARQ Verfahren interessant
- LDPC-Dekodierer erzeugt keine Bündelfehler am Ausgang
- LDPC für nahezu beliebige R und n konstruierbar
- „gute“ LDPC-Kodes übertreffen (bei großem n) die gegenwärtig bestbekanntesten Turbokodes
- LDPC-Kodes sind patentfrei

Anwendungen

- DVB-S2 Satellitenstandard (Verkettung von BCH und LDPC, BCH entfernt möglichen error-floor), 0,7 ... 1,2 dB von Grenze
- DÜ-Standards WLAN ($\leq 50 m$) und WiMAX ($\leq 50 km$)
- Verteilte Speichersysteme wie Cloud Computing, Behandlung von Auslöschungsbereichen (hybrid: Iterative Dekodierung/Gauß-Jordan-Elimination)
- IEEE 802.3an Standard für 10-Gigabit-Ethernet (Verkettung von CRC und (2048,1723)LDPC)
- Optische Netzwerke mit bit-flipping Algorithmus